

# Use Tools to Support Your Software Development

Sven Amann for BP16/17

# Let's Talk About

- Dependency Management
- Build Management
- Version Control / Configuration Management
- Continuous Integration

# Collaboration

- In software projects multiple developers work on the same codebase (in parallel)
- Changes need to be integrated and synchronized among team members (distribute changes to others, get changes from others)

# Dependencies

- 3rd-party dependencies are key to software development
- Managing libraries manually is error-prone
  - Where do I find `commons-io.jar`?
  - I have `libs/commons-io.jar`... is it 1.4 or 2.0?

# Build / Package

- Software needs to be built (compiled, tests run, binaries packaged, dependencies bundled, etc.)
- Doing this manually or through a bunch of arbitrary scripts is error-prone
  - Which environment do I need to run them?
  - What do I need to run when?

# Feedback

- Efficient work requires fast feedback
- Retrieving feedback manually is error-prone and will eventually be forgotten/skipped
  - Did my change break any part of the code?
  - Did I write enough tests?
  - Do things work in a production environment?

# Best Practices

- Use **Version Control** to synchronize/distribute changes within your team
- Use tools for **Dependency Management** to document dependencies and automate resolution
- Use tools for **Build Management** to automate the build and make it executable by every team member at all times

# Best Practices

- Use Version Control also for **Configuration Management**, i.e., to track you Dependency and Build Management.
- Use **Continuous Integration** to make sure the current code in the current configuration works and keeps working



# Let's Develop With

- **Eclipse** [tiny.cc/ld-eclipse](http://tiny.cc/ld-eclipse)
- **Maven** in Eclipse (m2e) [tiny.cc/ld-m2e](http://tiny.cc/ld-m2e)  
(Dependency and Build Management)
- **Git** in Eclipse (eGit) [tiny.cc/ld-egit](http://tiny.cc/ld-egit)  
(Version Control and Configuration Management)
- **Shippable**  
(Continuous Integration)

Live Demo

# LET'S DEVELOPER



[@letsdeveloper](https://youtube.com/letsdeveloper)